EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L13	52	(locking) near4 (access$5) near5 (resource semaphore) and L12	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 09:35
L12	25898	"710"/$.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 09:33
S10 1	710	710/116,117,57.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 09:31
L11	24	("4380798" or "4754398" or "4935866" or "4925311" or "5050070" or "5276886" or "5293491" or "5339443" or "5799195" or "5812876" or "5922057" or "5951662").pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 08:56
L10	16	(semaphore) and (arbitrat$5) same (resource) and L8	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 08:56
L9	38	(semaphore) and (arbitrat$5) same (resource) and L2	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 08:13
L3	52	(semaphore) and (arbitrat$5) same (priority) and L2	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 08:01
L8	50	(semaphore) near4 (inquir$5 check$5 decid$5) and L7	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 07:35
L6	46	(semaphore) near4 (inquir$5 check$5 decid$5) and L2	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 07:35
S13	96	(710/241.ccls. and @ad<"20001222") and((resource$3) and (shar$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/26 07:34
L7	5088	"710"/$.ccls. and ((resource$3) and (shar$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/26 07:34

EAST Search History

S105	324	(semaphore) and (arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/26 06:48
L2	12856	"712"/$.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/26 06:44
L1	12602	"712"/$.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/26 06:44
S109	3	(semaphore) and (arbitrat$5) same (priority) and S104	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:28
S108	16	(semaphore) and (arbitrat$5) same (priority) and S103	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:28
S107	3	(semaphore) and (arbitrat$5) same (priority) and S102	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:28
S106	9	(semaphore) and (arbitrat$5) same (priority) and S101	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:28
S98	91	(semaphore) near7 (processor) and (arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:22
S104	1648	714/9,12,31,10,11.ccls..	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:21
S103	1774	711/151,158,152.ccls.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:20
S97	3	(semaphore) near7 (specific) near5 (processor) and (arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:18

EAST Search History

S100	33	(semaphore) same(arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:14
S99	16	(semaphore) same (processor) same (arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 17:00
S96	0	(grant$5) near7 (semaphore) near7 (specific) near5 (processor) and (arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 16:42
S95	1	(allocat$5 assign$5 dedicat$5 access$5) near5 (semaphore) near7 (specific) near5 (processor) and (arbitrat$5) same (priority)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 16:41
S94	1	(allocat$5 assign$5 dedicat$5 access$5) near5 (semaphore) near7 (specific) near5 (processor) and (arbitrat$5) same (priority)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 16:40
S92	1	(allocat$5 assign$5 dedicat$5 access$5) near5 (semaphore) near7 (specific) near5 (processor) and S89	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 16:39
S93	6	("4402046" | "5274809" | "5377352" | "5455920" | "5485593" | "5754800").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2006/05/18 16:03
S91	0	(exclusive near5 (allocat$5 assign$5 dedicat$5 access$5)) near5 (semaphore) near7 (specific) near5 (processor) and S89	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 16:02
S90	2	(exclusive same (certain particular) same (processor) same (allocat$5 assign$5 dedicat$5 access$5)) near5 (semaphore) and S89	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 16:01
S89	27483	(multiprocessor near4 (die chip system))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 15:56
S86	5453	(exclusive near4 (aloocat$5 access$5))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 15:56
S88	12	("4604683" | "4630264" | "4642630" | "5067071" | "5187790" | "5237694" | "5261108" | "5276886" | "5289585" | "5339443" | "5446910" | "5485594").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2006/05/18 15:55
S87	22	S85 same S86	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 15:09

EAST Search History

S85	738	(semaphore) near4 (inquir$5 check$5 detect$5 determin$5 monitor$5)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	.OR	OFF	2006/05/18 15:08
S73	222	(semaphore) near4 (inquir$5 check$5 detect$5 determin$5 monitor$5) and (arbitrat$5)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 15:07
S84	2	"6282561".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 14:56
S83	10	(semaphore adj control adj (mechanism apparatus alogorithm))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 14:56
S82	3	(MSR (mode adj specific adj register)) and (semaphore adj control adj (mechanism apparatus alogorithm))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2006/05/18 14:54
S81	3	(MSR (mode adj specific adj register)) and (semaphore adj control adj (mechanism apparatus alogorithm))	US-PGPUB; USPAT; USOCR	OR	ON	2006/05/18 14:53
S80	7	("20020116438" | "5214652" | "5367668" | "5796937" | "6189117" | "6453427" | "6553512").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2006/05/18 14:52
S79	3	(semaphore near3 checker)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/18 14:46
S78	1	"5949974".pn.	US-PGPUB; USPAT; USOCR	OR	ON	2006/05/09 13:39
S77	16	("5696939" | "6018785" | "6023738" | "6047316" | "6065071" | "6081854" | "6134579" | "6195676" | "6199094" | "6243793" | "6260082" | "6282587" | "6292854" | "6343338" | "6519686" | "6567873").PN.	US-PGPUB; USPAT; USOCR	OR	ON	2006/05/09 13:39
S76	2	"6728238".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/09 12:58
S74	2	"6725457".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/09 12:58
S75	222	(semaphore) near4 (inquir$5 check$5 detect$5 determin$5 monitor$5) and (arbitrat$5)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/09 11:09
S5	91	710/220.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/05/09 10:52

EAST Search History

S72	3	(semaphore near3 checker)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/01/31 14:34
S58	2	"5872980".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/01/31 14:34
S70	2	"6148395".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/01/31 13:48
S69	2	"5418967".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2006/01/31 13:48
S68	20	710/240-244.ccls. and (shared) near5 (resource) same (arbitrat$5) same (robin)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/30 08:28
S67	194	710/240-244.ccls. and (arbitrat$5) same (robin)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/30 08:28
S66	7	(semaphore) same (arbitrat$5) same (robin)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/30 08:26
S65	0	710/240-244.ccls. and (semaphore) same (arbitrat$5) same (robin)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/30 08:26
S64	60	710/240-244.ccls. and (semaphore)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/30 08:25
S60	3	(semaphore) near4 (round-robin)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/30 08:18
S63	2	"6633910".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/22 16:27
S62	2	"5854901".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/22 16:27
S61	2	"6148395".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/22 16:10

EAST Search History

S59	7	(semaphore) near4 (round -robin)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/06/22 14:43
S4	36	(time) near15 (remain$5 left$5) near10 (clock window display) near10 (auction sale)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 17:02
S3	0	(generat$5 produc$5 deriv$5) near3 (time) near5 (remain$5 left$5) near10 (clock window display) near10 (auction sale)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 17:02
S2	264	(generat$5 produc$5 deriv$5) near3 (client slave secondary) near5 (clock time) near10 (server another main master) near10 (clock time) and (display show produc$5) near3 (clock time)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 17:00
S1	703	(generat$5 produc$5 deriv$5) near10 (client slave secondary) near5 (clock time) near10 (server another main master) near10 (clock time) and (display show produc$5) near3 (clock time)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 16:09
S57	2	("719"/$.ccls.) and (arbitrat$5) near5(shared common available system) near5(semaphore resource) near5(lock access)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 10:52
S56	32	("709"/$.ccls.) and (arbitrat$5) near5(shared common available system) near5(semaphore resource) near5(lock access)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 10:52
S55	175	("709"/$.ccls.) and (arbitrat$5 select$5 determin$5 detect$5) near5(shared common available system) near5(semaphore resource) near5(lock access)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 10:50
S54	70	("718"/$.ccls.) and (arbitrat$5 select$5 determin$5 detect$5) near5(shared common available system) near5(semaphore resource) near5(lock access)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 10:48
S53	290	("710"/$.ccls.) and (arbitrat$5 select$5 determin$5 detect$5) near5(shared common available system) near5(semaphore resource) near5(lock access)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 10:39
S52	423	("710"/$.ccls.) and (arbitrat$5 select$5 determin$5 detect$5) near10 (shared common available system) near10 (semaphore resource) near10 (lock access)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2004/10/19 09:20
S22	20	(((modif$4 edit$4 chang44) near10 (semaphore)) and @ad<"20001222") and (((acces$4)near10 (resourc$4)) and (Arbitra$5))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/10/19 09:16
S51	2	"5872980".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2004/05/13 09:06

EAST Search History

S50	41	(((multiprocessor and ((Single) adj (die chip))) and (Share$4 adj resource$4)) and @ad<"20001222") and (semaphore)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 11:09
S49	0	(((multiprocessor and ((Single) adj (die chip))) and (Share$4 adj resource$4)) and @ad<"20001222") and ((modif$4 edit$4 chang44) near10 (semaphore))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 11:08
S47	97	((multiprocessor and ((Single) adj (die chip))) and (Share$4 adj resource$4)) and @ad<"20001222"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 11:08
S48	0	(multiprocessor and ((Single) adj (die chip))) and (Share$4 adj resource$4) and ((modif$4 edit$4 chang44) near10 (semaphore))and @ad<"20001222"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 11:06
S46	122	(multiprocessor and ((Single) adj (die chip))) and (Share$4 adj resource$4)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 11:05
S45	1013	multiprocessor and ((Single) adj (die chip))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 10:27
S27	6832	Share$4 adj resource$4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 10:27
S44	21388	multiprocessor	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 10:26
S43	21978	multiprocessor$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 10:26
S36	2	"5592673".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/09 10:22
S42	2	"6477597".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 16:14
S41	2	"5872980".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 16:13
S29	2	"6212165".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 14:45

EAST Search History

S40	2	"6219763".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:31
S39	3	"9219763".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:31
S38	2	"4847755".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:31
S37	2	"4672536".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:28
S35	2	"5748969".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:24
S34	2	"5931924".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:23
S33	2	"6101584".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:22
S32	2	"5935234".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:22
S31	2	"6618778".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:20
S30	2	"5729702".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:19
S28	36	(Share$4 adj resource$4) and ((modif$4 edit$4 chang44) near10 (semaphore))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 13:16
S26	0	request adj ((modif$4 edit$4 chang44) near10 (semaphore))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 10:15
S21	115	((modif$4 edit$4 chang44) near10 (semaphore)) and @ad<"20001222"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 10:15

EAST Search History

S25	115	((modif$4 edit$4 chang44) near10 (semaphore)) and @ad<"20001222"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2003/12/08 10:14
S20	131	(modif$4 edit$4 chang44) near10 (semaphore)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 10:14
S24	2	"6009275".PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 10:00
S23	2	"5339443".PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 09:59
S18	51	Semaphore adj check$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/08 08:01
S19	2	Semaphore adj checker	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/03 17:08
S17	13	Semaphore adj modif$8	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/03 16:52
S16	9	((Acess$3 near5 Resource$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/03 16:42
S15	2	((Acess$3 near5 Resource$4))and ((semaphore near3 modi$10))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/03 16:30
S14	13	((710/241.ccls. and @ad<"20001222") and((resource$3) and (shar$4))) and(semaphore$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/03 16:27
S12	59644	L7and((resource$3) and (shar$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 16:26
S11	187	710/241.ccls. and @ad<"20001222"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 16:26
S10	99	710/241.ccls. and((resource$3) and (shar$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 16:26

EAST Search History

S9	196	710/241.ccls.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 16:26
S7	37	(710/220.ccls. and @ad<"20001222") and((resource$3) and (shar$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 16:24
S8	3	((710/220.ccls. and @ad<"20001222") and((resource$3) and (shar$4))) and semaphore	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 16:23
S6	82	710/220.ccls. and @ad<"20001222"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2003/12/02 15:38

Results for "(((semaphore<in>metadata) <and> (arbitrate<in>metadata))) <and> (pyr &..." e-mail
Your search matched **0** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((semaphore<in>metadata) <and> (arbitrate<in>metadata))) <and> (pyr >= 1950 Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan search.

Results for "(((semaphore<in>metadata) <and> (checker<in>metadata))) <and> (pyr >..." e-mail

Your search matched **0** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((semaphore<in>metadata) <and> (checker<in>metadata))) <and> (pyr >= 1950 Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan search.

Results for "(((semaphore<in>metadata) <and> (arbitration<in>metadata))) <and> (pyr..." e-mail
Your search matched **0** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((semaphore<in>metadata) <and> (arbitration<in>metadata))) <and> (pyr >= 195 Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan search.

Results for "(((semaphore<in>metadata) <and> (arbitrating<in>metadata))) <and> (pyr..." e-mail

Your search matched **0** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((semaphore<in>metadata) <and> (arbitrating<in>metadata))) <and> (pyr >= 195 Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan search.

Results for "(((semaphore<in>metadata) <and> (resource<in>metadata))) <and> (pyr &g..." e-mail
Your search matched **12** of **1351285** documents.
A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» Search Options

View Session History

New Search

Modify Search

(((semaphore<in>metadata) <and> (resource<in>metadata))) <and> (pyr >= 1950 Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

view selected items Select All Deselect All

1. **A generic simulator of real-time scheduling algorithms**
De Vroey, S.; Goossens, J.; Hernalsteen, C.;
Simulation Symposium, 1996. Proceedings of the 29th Annual
8-11 April 1996 Page(s):242 - 249
Digital Object Identifier 10.1109/SIMSYM.1996.492172
AbstractPlus | Full Text: PDF(748 KB) IEEE CNF
Rights and Permissions

2. **Semaphore queue priority assignment for real-time multiprocessor synch**
Lortz, V.B.; Shin, K.G.;
Software Engineering, IEEE Transactions on
Volume 21, Issue 10, Oct. 1995 Page(s):834 - 844
Digital Object Identifier 10.1109/32.469457
AbstractPlus | References | Full Text: PDF(1072 KB) IEEE JNL
Rights and Permissions

3. **Real-time synchronization between hard and soft tasks in RT-Linux**
Terrasa, A.; Garcia-Fornes, A.;
Real-Time Computing Systems and Applications, 1999. RTCSA '99. Sixth Inte
Conference on
13-15 Dec. 1999 Page(s):434 - 441
Digital Object Identifier 10.1109/RTCSA.1999.811295
AbstractPlus | Full Text: PDF(640 KB) IEEE CNF
Rights and Permissions

4. **Petri net models of process synchronization mechanisms**
Zuberek, W.M.;
Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceeding
International Conference on
Volume 1, 12-15 Oct. 1999 Page(s):841 - 847 vol.1
Digital Object Identifier 10.1109/ICSMC.1999.814201
AbstractPlus | Full Text: PDF(560 KB) IEEE CNF
Rights and Permissions

5. **Scheduling complex real-time tasks in an embedded distributed system**
Maillet, L.; Fraboul, C.;
Real-Time Systems, 1995. Proceedings., Seventh Euromicro Workshop on
14-16 June 1995 Page(s):62 - 65

Digital Object Identifier 10.1109/EMWRTS.1995.514293

AbstractPlus | Full Text: PDF(328 KB) IEEE CNF
Rights and Permissions

6. **A RISC development case study**
Cates, R.L.; Farrell, J.J., III;
Compcon Spring '90. 'Intellectual Leverage', Digest of Papers. Thirty-Fifth IEEE Society International Conference.
26 Feb.-2 March 1990 Page(s):290 - 294
Digital Object Identifier 10.1109/CMPCON.1990.63689

AbstractPlus | Full Text: PDF(240 KB) IEEE CNF
Rights and Permissions

7. **A stack-based resource allocation policy for realtime processes**
Baker, T.P.;
Real-Time Systems Symposium, 1990. Proceedings., 11th
5-7 Dec. 1990 Page(s):191 - 200
Digital Object Identifier 10.1109/REAL.1990.128747

AbstractPlus | Full Text: PDF(764 KB) IEEE CNF
Rights and Permissions

8. **A class of multiprocessor architectures for real-time DSP**
Bier, J.C.; Lee, E.A.;
Circuits and Systems, 1990., IEEE International Symposium on
1-3 May 1990 Page(s):2622 - 2625 vol.4
Digital Object Identifier 10.1109/ISCAS.1990.112546

AbstractPlus | Full Text: PDF(416 KB) IEEE CNF
Rights and Permissions

9. **Recoverable user-level mutual exclusion**
Bohannon, P.; Lieuwen, D.; Silberschatz, A.; Sudarshan, S.; Gava, J.;
Parallel and Distributed Processing, 1995. Proceedings. Seventh IEEE Sympo
25-28 Oct. 1995 Page(s):293 - 301
Digital Object Identifier 10.1109/SPDP.1995.530698

AbstractPlus | Full Text: PDF(816 KB) IEEE CNF
Rights and Permissions

10. **Resource binding-a universal approach to parallel programming**
Shing, H.; Ni, L.M.;
Supercomputing '90. Proceedings of
12-16 Nov. 1990 Page(s):574 - 583
Digital Object Identifier 10.1109/SUPERC.1990.130072

AbstractPlus | Full Text: PDF(748 KB) IEEE CNF
Rights and Permissions

11. **A distributed algorithm for parameterized semaphore operations**
Nelakonda, M.; Mizuno, M.; Neilsen, M.L.;
Computing and Information, 1992. Proceedings, ICCI '92., Fourth International
28-30 May 1992 Page(s):466 - 469
Digital Object Identifier 10.1109/ICCI.1992.227611

AbstractPlus | Full Text: PDF(268 KB) IEEE CNF
Rights and Permissions

12. **Enabling technologies for e-business**
Peacock, R.;
IT Professional
Volume 2, Issue 4, July-Aug. 2000 Page(s):52 - 54
Digital Object Identifier 10.1109/6294.869385

AbstractPlus | Full Text: PDF(108 KB) IEEE JNL
Rights and Permissions

Published before January 2001
Terms used
semaphore resource check arbitrating arbitration arbitrate

Found **530** of **116,808**

Sort results by: relevance
Display results: expanded form

Save results to a Binder
Search Tips
Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next
Best 200 shown Relevance scale

1 Synchronization in actor systems
Russell Atkinson, Carl Hewitt
January 1977 **Proceedings of the 4th ACM SIGACT-SIGPLAN symposium on Principles of programming languages**
Publisher: ACM Press
Full text available: pdf(1.13 MB) Additional Information: full citation, abstract, references, citings

This paper presents a mechanism for the arbitration of parallel requests to shared resources. This mechanism is the serialize, which may be described as a kind of protection mechanism, in that it prevents improper orders of access to a protected resource. The mechanism is a generalization and improvement of the monitor mechanism of Brinch-Hansen and Hoare.Serializers attempt to systematize and abstract desirable structural features of synchronization control structure into a coherent language co ...

2 A survey of commercial parallel processors
Edward Gehringer, Janne Abullarade, Michael H. Gulyn
September 1988 **ACM SIGARCH Computer Architecture News**, Volume 16 Issue 4
Publisher: ACM Press
Full text available: pdf(2.96 MB) Additional Information: full citation, abstract, citings, index terms

This paper compares eight commercial parallel processors along several dimensions. The processors include four shared-bus multiprocessors (the Encore Multimax, the Sequent Balance system, the Alliant FX series, and the ELXSI System 6400) and four network multiprocessors (the BBN Butterfly, the NCUBE, the Intel iPSC/2, and the FPS T Series). The paper contrasts the computers from the standpoint of interconnection structures, memory configurations, and interprocessor communication. Also, the share ...

3 Are crossbars really dead?: the case for optical multiprocessor interconnect systems
Andreas G. Nowatzyk, Paul R. Prucnal
May 1995 **ACM SIGARCH Computer Architecture News , Proceedings of the 22nd annual international symposium on Computer architecture ISCA '95**, Volume 23 Issue 2
Publisher: ACM Press
Full text available: pdf(1.16 MB) Additional Information: full citation, abstract, references, citings, index terms

Crossbar switches are rarely considered for large, scalable multiprocessor interconnect

systems because they require $O(n^2)$ switching elements, are difficult to control efficiently and are hard to implement once their size becomes too large to fit on one integrated circuit. However these problems are technology dependent and a recent innovation in fiber optic devices has led to a new implementation of crossbar switches that does not share these problems while retaining the full advanta ...

4 The design and development of a very high speed system bus—the encore Mutlimax nanobus

David J. Schanin

November 1986 **Proceedings of 1986 ACM Fall joint computer conference**

Publisher: IEEE Computer Society Press

Full text available: pdf(910.65 KB) Additional Information: full citation, references, citings, index terms

5 Processor coupling: integrating compile time and runtime scheduling for parallelism

Stephem W. Keckler, William J. Dally

April 1992 **ACM SIGARCH Computer Architecture News , Proceedings of the 19th annual international symposium on Computer architecture ISCA '92**, Volume 20 Issue 2

Publisher: ACM Press

Full text available: pdf(1.32 MB) Additional Information: full citation, abstract, references, citings, index terms

The technology to implement a single-chip node composed of 4 high-performance floating-point ALUs will be available by 1995. This paper presents processor coupling, a mechanism for controlling multiple ALUs to exploit both instruction-level and inter-thread parallelism, by using compile time and runtime scheduling. The compiler statically schedules individual threads to discover available intra-thread instruction-level parallelism. The runtime scheduling mechanism interleaves threads, explo ...

6 Experience Using Multiprocessor Systems—A Status Report

Anita K. Jones, Peter Schwarz

June 1980 **ACM Computing Surveys (CSUR)**, Volume 12 Issue 2

Publisher: ACM Press

Full text available: pdf(4.48 MB) Additional Information: full citation, references, citings, index terms

7 Monitor classification

Peter A. Buhr, Michel Fortier, Michael H. Coffin

March 1995 **ACM Computing Surveys (CSUR)**, Volume 27 Issue 1

Publisher: ACM Press

Full text available: pdf(3.41 MB) Additional Information: full citation, abstract, references, citings, index terms

One of the most natural, elegant, and efficient mechanisms for synchronization and communication, especially for systems with shared memory, is the monitor. Over the past twenty years many kinds of monitors have been proposed and implemented, and many modern programming languages provide some form of monitor for concurrency control. This paper presents a taxonomy of monitors that encompasses all the extant monitors and suggests others not found in the literature or in exist ...

Keywords: classification, monitors

8

The directory-based cache coherence protocol for the DASH multiprocessor
Daniel Lenoski, James Laudon, Kourosh Gharachorloo, Anoop Gupta, John Hennessy
May 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 17th annual international symposium on Computer Architecture ISCA '90**, Volume 18 Issue 3a
Publisher: ACM Press
Full text available: pdf(1.74 MB) Additional Information: full citation, abstract, references, citings, index terms

DASH is a scalable shared-memory multiprocessor currently being developed at Stanford's Computer Systems Laboratory. The architecture consists of powerful processing nodes, each with a portion of the shared-memory, connected to a scalable interconnection network. A key feature of DASH is its distributed directory-based cache coherence protocol. Unlike traditional snoopy coherence protocols, the DASH protocol does not rely on broadcast; instead it uses point-to-point messages sent between th ...

9 Distributed FIFO allocation of identical resources using small shared space
Michael J. Fischer, Nancy A. Lynch, James E. Burns, Allan Borodin
January 1989 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 11 Issue 1
Publisher: ACM Press
Full text available: pdf(1.80 MB) Additional Information: full citation, abstract, references, citings, index terms, review

We present a simple and efficient algorithm for the FIFO allocation of k identical resources among asynchronous processes that communicate via shared memory. The algorithm simulates a shared queue but uses exponentially fewer shared memory values, resulting in practical savings of time and space as well as program complexity. The algorithm is robust against process failure through unannounced stopping, making it attractive also for use in an environment of processes of wide ...

10 Characteristics of scalability and their impact on performance
André B. Bondi
September 2000 **Proceedings of the 2nd international workshop on Software and performance WOSP '00**
Publisher: ACM Press
Full text available: pdf(168.62 KB) Additional Information: full citation, references, index terms

Keywords: load scalability, performance, structural and space scalability

11 Lessons learned from the OS/400 OO project
William Berg, Marshall Cline, Mike Girou
October 1995 **Communications of the ACM**, Volume 38 Issue 10
Publisher: ACM Press
Full text available: pdf(339.92 KB) Additional Information: full citation, abstract, references, citings, index terms

This article describes some of the lessons learned when a team of 150 developers with a minimal prior exposure to object-oriented (OO) technology undertook a large development project. Team members became proficient in OO design, using C++ as an OO language rather than just using C++ as a better C, and developed IBM's RISC version of the AS/400 and System/36 operating systems from 1992 to 1994 in Rochester, Minnesota. The project contains 14,000 thousand classes, 90,000 thousand methods, an ...

12 Protection in the Hydra Operating System

Ellis Cohen, David Jefferson

November 1975 **Proceedings of the fifth ACM symposium on Operating systems principles**

Publisher: ACM Press

Full text available: pdf(1.98 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes the capability based protection mechanisms provided by the Hydra Operating System Kernel. These mechanisms support the construction of user-defined protected subsystems, including file and directory subsystems, which do not therefore need to be supplied directly by Hydra. In addition, we discuss a number of well known protection problems, including Mutual Suspicion, Confinement and Revocation, and we present the mechanisms that Hydra supplies in order to solve them.

Keywords: Capability, Confinement, Mutual suspicion, Operating system, Protected subsystem, Protection, Protection problem, Revocation, Type

13 Policy/mechanism separation in Hydra

R. Levin, E. Cohen, W. Corwin, F. Pollack, W. Wulf

November 1975 **Proceedings of the fifth ACM symposium on Operating systems principles**

Publisher: ACM Press

Full text available: pdf(988.09 KB) Additional Information: full citation, abstract, references, citings, index terms

The extent to which resource allocation policies are entrusted to user-level software determines in large part the degree of flexibility present in an operating system. In Hydra the determination to separate mechanism and policy is established as a basic design principle and is implemented by the construction of a kernel composed (almost) entirely of mechanisms. This paper presents three such mechanisms (scheduling, paging, protection) and examines how external policies which manipulate the ...

Keywords: Mechanism, Operating system, Paging, Policy, Protection, Resource allocation, Scheduling

14 The gould NP1 system interconnecting

D. J. Vianney, J. H. Thomas, V. Rabaza

June 1988 **Proceedings of the 2nd international conference on Supercomputing**

Publisher: ACM Press

Full text available: pdf(1.28 MB) Additional Information: full citation, abstract, references, index terms

The Gould NP1 is a multicomputer multiprocessing system designed for high performance and parallel processing required in diverse scientific and engineering applications. The NP1's basic building block is a dual-processor single bus system which can be expanded up to eight processors over four system buses. This paper discusses the overall design and implementation of tne NP1 system interconnection in particularly the inter-system bus link which interconnects four system buses to ...

15 Third Generation Computer Systems

Peter J. Denning

December 1971 **ACM Computing Surveys (CSUR)**, Volume 3 Issue 4

Publisher: ACM Press

Full text available: pdf(3.52 MB) Additional Information: full citation, abstract, references, citings, index terms

The common features of third generation operating systems are surveyed from a general view, with emphasis on the common abstractions that constitute at least the basis for a "theory" of operating systems. Properties of specific systems are not discussed except where examples are useful. The technical aspects of issues and concepts are stressed, the nontechnical aspects mentioned only briefly. A perfunctory knowledge of third generation systems is presumed.

16 Monit: a performance monitoring tool for parallel and pseudo-parallel programs

Teemu Kerola, Herb Schwetman

May 1987 **ACM SIGMETRICS Performance Evaluation Review , Proceedings of the 1987 ACM SIGMETRICS conference on Measurement and modeling of computer systems SIGMETRICS '87**, Volume 15 Issue 1

Publisher: ACM Press

Full text available: pdf(969.76 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes a performance monitoring system, Monit, developed for performance evaluation of parallel systems. Monit uses trace files that are generated during the execution of parallel programs. Monit analyzes these trace files and produces time-oriented graphs of resource usage and system queues. Users interactively select the displayed items, resolution, and time intervals of interest. The current implementation of Monit is for SUN-3 workstation, but the program is easily adaptab ...

17 Data management requirements: The similarity of memory management, database systems, and message processing

Olin H. Bray

January 1977 **ACM SIGIR Forum , ACM SIGARCH Computer Architecture News , ACM SIGMOD Record , Proceedings of the 3rd workshop on Computer architecture : Non-numeric processing**, Volume 12 , 6 , 9 Issue 1 , 2 , 2

Publisher: ACM Press

Full text available: pdf(927.65 KB) Additional Information: full citation, abstract, references, index terms

Memory management, database management, and message processing have in the past been defined in a relatively narrow way. With memory management the problem was to obtain cost effective use of real memory. Given a multiprogrammed environment, virtual memory systems allowed more effective use of expensive real memory. Memory management has become even more important with the development of very large and complex memory hierarchies. Database management systems were developed to allow the more ...

18 The rendezvous is dead—long live the protected object

Dragan Macos, Frank Mueller

November 1998 **ACM SIGAda Ada Letters , Proceedings of the 1998 annual ACM SIGAda international conference on Ada SIGAda '98**, Volume XVIII Issue 6

Publisher: ACM Press

Full text available: pdf(750.79 KB) Additional Information: full citation, references, index terms

19 Towards a taxonomy of software connectors

Nikunj R. Mehta, Nenad Medvidovic, Sandeep Phadke

June 2000 **Proceedings of the 22nd international conference on Software engineering**

Publisher: ACM Press

Full text available: pdf(184.27 KB) Additional Information: full citation, abstract, references, citings, index terms

Software systems of today are frequently composed from prefabricated, heterogeneous components that provide complex functionality and engage in complex interactions. Existing research on component-based development has mostly focused on component structure, interfaces, and functionality. Recently, software architecture has emerged as an area that also places significant importance on component interactions, embodied in the notion of software connectors. However, the current level of underst ...

Keywords: classification, software architecture, software connector, taxonomy

20 OMP: a RISC-based multiprocessor using orthogonal-access memories and multiple spanning buses

K. Hwang, M. Dubois, D. K. Panda, S. Rao, S. Shang, A. Uresin, W. Mao, H. Nair, M. Lytwyn, F. Hsieh, J. Liu, S. Mehrotra, C. M. Cheng

June 1990 **ACM SIGARCH Computer Architecture News , Proceedings of the 4th international conference on Supercomputing ICS '90**, Volume 18 Issue 3b
Publisher: ACM Press

This paper presents the architectural design and RISC based implementation of a prototype supercomputer, namely the Orthogonal MultiProcessor (OMP). The OMP system is constructed with 16 Intel 1860 RISC microprocessors and 256 parallel memory modules, which are 2-D interleaved and orthogonally accessed using custom-designed spanning buses. The architectural design has been validated by a CSIM-based multiprocessor simulator. The design choices are based on worst-case delay a ...

Terms used **semaphore checker** Found **76** of **176,279**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 76 Result page: **1** 2 3 4 next

Relevance scale

1 Making operating systems more robust: RacerX: effective, static detection of race conditions and deadlocks

Dawson Engler, Ken Ashcraft

October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**

Publisher: ACM Press

Full text available: pdf(310.63 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper describes RacerX, a static tool that uses flow-sensitive, interprocedural analysis to detect both race conditions and deadlocks. It is explicitly designed to find errors in large, complex multithreaded systems. It aggressively infers checking information such as which locks protect which operations, which code contexts are multithreaded, and which shared accesses are dangerous. It tracks a set of code features which it uses to sort errors both from most to least severe. It uses novel ...

Keywords: deadlock detection, program checking, race detection

2 Scalable error detection using boolean satisfiability

Yichen Xie, Alex Aiken

January 2005 **ACM SIGPLAN Notices , Proceedings of the 32nd ACM SIGPLAN-SIGACT symposium on Principles of programming languages POPL '05**, Volume 40 Issue 1

Publisher: ACM Press

Full text available: pdf(276.56 KB) Additional Information: full citation, abstract, references, citings, index terms

We describe a software error-detection tool that exploits recent advances in boolean satisfiability (SAT) solvers. Our analysis is path sensitive, precise down to the bit level, and models pointers and heap data. Our approach is also highly scalable, which we achieve using two techniques. First, for each program function, several optimizations compress the size of the boolean formulas that model the control- and data-flow and the heap locations accessed by a function. Second, summaries in the sp ...

Keywords: boolean satisfiability, error detection, program analysis

3 Automating comprehensive safety analysis of concurrent programs using verisoft and

TXL
Juergen Dingel, Hongzhi Liang
October 2004 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 12th ACM SIGSOFT twelfth international symposium on Foundations of software engineering SIGSOFT '04/FSE-12**, Volume 29 Issue 6
Publisher: ACM Press
Full text available: pdf(188.82 KB) Additional Information: full citation, abstract, references, index terms

In run-time safety analysis the executions of a concurrent program are monitored and analyzed with respect to safety properties. Similar to testing, run-time analysis is quite efficient, but it also tends to be incomplete. The results pertain only to the observed executions which may constitute just a small subset of all possible executions.

In this paper, we describe a tool called ViP which uses the software model checker VeriSoft to perform comprehensive run-time safety analyses of c ...

Keywords: TXL, VeriSoft, past-time linear temporal logic, run-time monitoring, safety analysis, software model checking, source code transformation

4 Re-usable software design for programmable logic controllers
Flavio Bonfatti, Gianni Gadda, Paola Daniela Monari
November 1995 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 1995 workshop on Languages, compilers, & tools for real-time systems LCTES '95**, Volume 30 Issue 11
Publisher: ACM Press
Full text available: pdf(955.71 KB) Additional Information: full citation, abstract, references, citings, index terms

It is the aim of this paper to present a model conceived for supporting the initial, critical phases of PLC software life cycle, namely requirement specification, requirement analysis and software design, to obtain reusable code. The model, named EASIER, is based on an object-oriented paradigm where the message-method mechanism is replaced by the law and action primitives, since they are more suitable to cope with the real-time, cyclic nature of PLC software. Software re-usability is pursued by ...

5 A model parametric real-time logic
Angelo Morzenti, Dino Mandrioli, Carlo Ghezzi
October 1992 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 14 Issue 4
Publisher: ACM Press
Full text available: pdf(3.54 MB) Additional Information: full citation, abstract, references, citings, index terms, review

TRIO is a formal notation for the logic-based specification of real-time systems. In this paper the language and its straightforward model-theoretic semantics are briefly summarized. Then the need for assigning a consistent meaning to TRIO specifications is discussed, with reference to a variety of underlying time structures such as infinite-time structures (both dense and discrete) and finite-time structures. The main motivation is the ability to validate formal specifications. A solution ...

Keywords: first-order logic, formal specifications, model-theoretic semantics, real-time systems, requirements validation, temporal logic

6 A provably correct operating system: &dgr;-core
Ming-Yuan Zhu, Lei Luo, Guang-Zhe Xiong

January 2001 **ACM SIGOPS Operating Systems Review**, Volume 35 Issue 1
Publisher: ACM Press
Full text available: pdf(812.24 KB) Additional Information: full citation, abstract, citings, index terms

Embedded and real-time applications are appearing in a wider variety and in more critical applications than ever before, and they are generating more revenue from the services they provide. At the same time, these applications are growing larger and more complex as the market calls for higher performance and reduced costs. To meet market demands and reduce the risks inherent in building complex systems, manufacturers must find and apply technologies that help reach their goals and minimize risks ...

7 Technical papers: concurrency: Software model checking in practice: an industrial case study
Satish Chandra, Patrice Godefroid, Christopher Palm
May 2002 **Proceedings of the 24th International Conference on Software Engineering**
Publisher: ACM Press
Full text available: pdf(1.16 MB) Additional Information: full citation, abstract, references, citings, index terms

We present an application of software model checking to the analysis of a large industrial software product: Lucent Technologies' CDMA call-processing library. This software is deployed on thousands of base stations in wireless networks world-wide, where it sets up and manages millions of calls to and from mobile devices everyday. Our analysis of this software was carried out using VeriSoft, a tool developed at Bell Laboratories that implements model-checking algorithms for systematically testin ...

8 A system and language for building system-specific, static analyses
Seth Hallem, Benjamin Chelf, Yichen Xie, Dawson Engler
May 2002 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2002 Conference on Programming language design and implementation PLDI '02**, Volume 37 Issue 5
Publisher: ACM Press
Full text available: pdf(276.96 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents a novel approach to bug-finding analysis and an implementation of that approach. Our goal is to find as many serious bugs as possible. To do so, we designed a flexible, easy-to-use extension language for specifying analyses and an efficent algorithm for executing these extensions. The language, *metal*, allows the users of our system to specify a broad class of analyses in terms that resemble the intuitive description of the rules that they check. The system, *xgcc* ...

Keywords: error detection, extensible compilation

9 A newcomer's impressions of Scheme
Gregory V. Wilson
January 1994 **ACM SIGPLAN Lisp Pointers**, Volume VII Issue 1-2
Publisher: ACM Press
Full text available: pdf(356.13 KB) Additional Information: full citation, abstract, index terms

In July and November of 1993 I posted a summary of my first impressions of Scheme to the Internet news group comp.lang.scheme. The articles engendered a great deal of follow-up, from which I learned a great deal more about the language. I am grateful to everyone who commented on the original postings, and hope that this article will stimulate as much discussion as they did.

10 A multiple stream microprocessor prototype system: AMP-1

Edward S. Davidson

May 1980 **Proceedings of the 7th annual symposium on Computer Architecture**

Publisher: ACM Press

Full text available: pdf(698.01 KB) Additional Information: full citation, abstract, references, citings, index terms

A general-purpose multiple-stream processor with shared memory and a single time-multiplexed synchronous bus has been implemented. The AMP-1 system uses eight standard microprocessors and 64K bytes of memory. The design is highly efficient in the use of processor, bus, and memory resources. Preliminary performance measurements agree closely with an analytic memory access conflict model and show extremely low conflict-based performance degradation. Heavy interleaving of the memory and effect ...

11 Single sourcing: Authoring translation-ready documents: is software the answer?

Jennifer Wells Akis, Stephanie Brucker, Virginia Chapman, Layne Ethington, Bob Kuhns, PJ Schemenaur

October 2003 **Proceedings of the 21st annual international conference on Documentation**

Publisher: ACM Press

Full text available: pdf(203.10 KB) Additional Information: full citation, abstract, references, index terms

This paper describes the experiences of a publications group and a localization group at Sun Microsystems, Inc., of evaluating and then adopting a translatability checker. Translatability checkers are fairly new to Sun. This paper's findings are restricted to the company's experience, rather than to the long-term effects of translatability checkers on publications departments. This paper attempts to follow these guidelines for translatability checking:

- Sentences contain no more than 25 wo ...

Keywords: style, sun proof, translatability

12 Modeling methodology b: Simulation and verification I: from simulation to verification (and back)

Harald Rueß, Leonardo de Moura

December 2003 **Proceedings of the 35th conference on Winter simulation: driving innovation**

Publisher: Winter Simulation Conference

Full text available: pdf(198.00 KB) Additional Information: full citation, abstract, references

Symbolic evaluation is the execution of software and software designs on inputs given as symbolic or explicit constants along with constraints on these inputs. Efficient symbolic evaluation is now feasible due to recent advances in efficient decision procedures and symbolic model checking. Symbolic evaluation can be applied to partially implemented descriptions and provides wider coverage and greater assurance than testing and traditional simulation alone. Unlike full formal verification, sym ...

13 Using a coordination language to specify and analyze systems containing mobile components

P. Ciancarini, F. Franzé, C. Mascolo

April 2000 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 9 Issue 2

Publisher: ACM Press

Full text available: pdf(306.92 KB) Additional Information: full citation, abstract, references, citings, index terms, review

New computing paradigms for network-aware applications need specification languages able to deal with the features of mobile code-based systems. A coordination language provides a formal framework in which the interaction of active entities can be expressed. A coordination language deals with the creation and destruction of code or complex agents, their communication activites, as well as their distribution and mobility in space. We show how the coordination language PoliS offers a flexible ...

14 New modeling approaches and their application: Exploiting polymorphism in HW design: a case study in the ATM domain

Luigi Pomante

September 2004 **Proceedings of the 2nd IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis**

Publisher: ACM Press

Full text available: pdf(264.69 KB) Additional Information: full citation, abstract, references, index terms

The need of raising the level of abstraction and improving reuse in HW design suggests the adoption of an object-oriented (OO) design methodology based on *SystemC-Plus* (i.e. an enhanced *SystemC*). Such a methodology, developed during the *ODETTE* IST project, allows the exploitation of the key features of the OO paradigm (i.e. information hiding, inheritance, and polymorphism) at the behavioral level of description while guaranteeing synthesizability. In this context, the goal ...

Keywords: ATM, SystemC, design, polymorphism, synthesis

15 Verifying linear time temporal logic properties of concurrent Ada programs with quasar

S. Evangelista, C. Kaiser, J. F. Pradat-Peyre, P. Rousseau

December 2003 **ACM SIGAda Ada Letters , Proceedings of the 2003 annual ACM SIGAda international conference on Ada: the engineering of correct and reliable software for real-time & distributed systems using ada and related technologies SigAda '03**, Volume XXIV Issue 1

Publisher: ACM Press

Full text available: pdf(171.96 KB) Additional Information: full citation, abstract, references, index terms

In this paper we present an original and useful way for specifying and verifying temporal properties of concurrent programs with our tool named Quasar. Quasar is based on ASIS and uses formal methods (model checking). Properties that can be checked are either general, like deadlock or fairness, or more context specific, referring to tasks states or to value of variables; properties are then expressed in temporal logic. In order to simplify the expression of these properties, we define some templ ...

Keywords: concurrency, petri nets, software verification, temporal logic

16 Specification, verification, and synthesis of concurrency control components

Tuba Yavuz-Kahveci, Tevfik Bultan

July 2002 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 2002 ACM SIGSOFT international symposium on Software testing and analysis ISSTA '02**, Volume 27 Issue 4

Publisher: ACM Press

Full text available: pdf(315.84 KB) Additional Information: full citation, abstract, references, citings

Run-time errors in concurrent programs are generally due to the wrong usage of synchronization primitives such as monitors. Conventional validation techniques such as testing become ineffective for concurrent programs since the state space increases exponentially with the number of concurrent processes. In this paper, we propose an

approach in which 1) the concurrency control component of a concurrent program is formally specified, 2) it is verified automatically using model checking, and 3) the ...

Keywords: concurrent programming, infinite-state model checking, monitors, specification languages

17 Applications of model checking at Honeywell Laboratories

Darren Cofer, Eric Engstrom, Robert Goldman, David Musliner, Steve Vestal

May 2001 **Proceedings of the 8th international SPIN workshop on Model checking of software**

Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(93.14 KB) Additional Information: full citation, abstract, references

This paper provides a brief overview of five projects in which Honeywell has successfully used or developed model checking methods in the verification and synthesis of safety-critical systems.

18 A High-performance, memory-based interconnection system for multicomputer environments

Creve Maples

November 1990 **Proceedings of the 1990 ACM/IEEE conference on Supercomputing**

Publisher: IEEE Computer Society

Full text available: pdf(1.70 MB) Additional Information: full citation, abstract, references

The objective of this paper is to outline the design and operation of a very high-performance, memory-mapped interconnection system, called Merlin. The design can be effectively utilized to interconnect processors in a wide variety on environments, ranging from closely-coupled, dedicated systems to distributed workstations. The system provides a uniform approach to parallel programming which is independent of interconnection topology, processing elements, and languages. By using dynamically mapp ...

19 Ownership types for safe programming: preventing data races and deadlocks

Chandrasekhar Boyapati, Robert Lee, Martin Rinard

November 2002 **ACM SIGPLAN Notices , Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications OOPSLA '02**, Volume 37 Issue 11

Publisher: ACM Press

Full text available: pdf(459.57 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents a new static type system for multithreaded programs; well-typed programs in our system are guaranteed to be free of data races and deadlocks. Our type system allows programmers to partition the locks into a fixed number of equivalence classes and specify a partial order among the equivalence classes. The type checker then statically verifies that whenever a thread holds more than one lock, the thread acquires the locks in the descending order.Our system also allows programmer ...

Keywords: data races, deadlocks, encapsulation, ownership types

20 The current state of proving programs correct

Ralph L. London

August 1972 **Proceedings of the ACM annual conference - Volume 1**

Publisher: ACM Press

Full text available: pdf(471.62 KB) Additional Information: full citation, abstract, references, citings, index terms

Presented are successful efforts in proving that computer programs are correct. Included are (i) the methods used, (ii) the wide class of programs (including systems programs) that have been proved, and (iii) implemented computer systems for demonstrating correctness. There is also a partially annotated bibliography.

Keywords: Algorithm proof, Assertions, Correctness, Inductive assertions, Program proving, Proof, Proof checker, Proof methods, Reliability, Software reliability, Survey, Tutorial, Verification, Verifier

Results 1 - 20 of 76 Result page: **1** 2 3 4 next

Results for "(semaphore checker<in>metadata)" e-mail
Your search matched 0 documents.
A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

» Search Options

View Session History

New Search

Modify Search

(semaphore checker<in>metadata) Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

No results were found.

Please edit your search criteria and try again. Refer to the Help pages if you need assistan search.